

**Dilema**

Estamos en condiciones de crear las bases de datos, pero hay una duda que me atormenta. Par los atributos en el diseño de nuestra base de datos hemos utilizados muchas veces abreviaturas o acrónimos otras veces no.

## Creación de la base SGE

```
MariaDB [(none)]> CREATE DATABASE sge;
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [(none)]>
```

## Ponemos en uso la base SGE

```
MariaDB [(none)]> USE sge;
Database changed
MariaDB [sge]>
```

Comprobamos que nuestra base de datos aún está vacía

```
MariaDB [sge]> SHOW TABLES;
Empty set (0.000 sec)
```

```
MariaDB [sge]>
```

## Creación de las Tablas

### EspacioCurricular

Al final hay algunos cambios que haremos en los nombres de algunos atributos por eso pondremos primero la tabla con la instrucción original y luego con los cambios. Utilizaremos el segundo de éstos.

```
EspacioCurricular(id_ec, esp_c, curso, div, nivel, ori, f_cre, cant_h, res_cre, cont)
```

```
EspacioCurricular(id_ec, esp_c, curso, division, nivel, orientacion, fecha_cre, carga_h,
resol_cre, contenido)
```

Al momento de crear las tablas debemos elegir el tipo de datos. El siguiente cuadro muestra los tipos de datos elegidos para cada campo de nuestra tabla EspacioCurricular

#### Descripción

EspacioCurricular		
Nombre	Tipo	Descripción
id_ec	INT	PK alias pk_cec
esp_c	VARCHAR(80)	Programación Orientada a Objetos
curso	CHAR(1)	1, 2, etc.
division	CHAR(1)	A, B, C, etc.
nivel	CHAR(3)	ESB o ESO
orientacion	VARCHAR(50)	Informática
fecha_cre	DATETIME	2013
carga_h	TINYINT	Se trata de <b>horas Cátedra Semanal</b>
resol_cre	VARCHAR	
contenido	VARCHAR(9)	Orientado

#### alias de claves

pk\_cec alias de id\_ec (primary key) entidad EspacioCurricular

```
MariaDB [sge]> CREATE TABLE EspacioCurricular(
-> id_ec INT,
-> nombre_ec VARCHAR(150),
-> curso CHAR(1),
-> division CHAR(1),
-> nivel CHAR(3),
-> orientacion VARCHAR(150),
-> fecha_cre DATETIME,
-> carga_h TINYINT,
-> resol_cre VARCHAR(15),
-> contenido VARCHAR(9),
-> CONSTRAINT pk_cec PRIMARY KEY (id_ec));
Query OK, 0 rows affected, 1 warning (0.004 sec)
```

MariaDB [sge]>

## Provincia

Provincia(id\_prov, prov, cp\_prov)  
 Provincia(id\_prov, provincia, cp\_prov)

Provincia		
Nombre	Tipo	Descripción
id_prov	INT	PK alias pk_cp con CONSTRAINT
provincia	VARCHAR(50)	
cp_prov	CHAR(2)	Código Postal de Provincia Compuesto Por Dos Letras.

### alias de claves

pk\_cp alias de id\_prov (primary key) entidad Provincia

```
MariaDB [sge]> CREATE TABLE Provincia(
-> id_prov INT,
-> provincia VARCHAR(50),
-> cp_prov CHAR(2),
-> CONSTRAINT pk_cp PRIMARY KEY (id_prov));
Query OK, 0 rows affected, 1 warning (0.004 sec)
```

MariaDB [sge]>

Voy hasta aquí \_\_\_\_\_

## Localidad

Localidad(id\_loc, id\_prov2, loc, cp\_loc)  
 Localidad(id\_loc, id\_prov2, localidad, cp\_loc)

Localidad		
Nombre	Tipo	Descripción
id_loc	INT	PK alias pk_cl
id_cprov2	INT	FK alias fk_cprov2
localidad	VARCHAR(50)	
cp_loc	CHAR(4)	Código postal de la localidad compuesto por 4 dígitos.

### alias de claves

pk\_cl alias de id\_loc primary key entidad Localidad  
 fk\_cprov2 alias de id\_prov2 foreign key entidad Provincia

```
MariaDB [sge]> CREATE TABLE Localidad(
-> id_loc INT,
-> id_cprov2 INT,
-> localidad VARCHAR(50),
-> cp_loc CHAR(4),
-> CONSTRAINT pk_cl PRIMARY KEY (id_loc),
-> CONSTRAINT fk_cprov2 FOREIGN KEY (id_cprov2) REFERENCES Provincia(id_prov));
Query OK, 0 rows affected, 1 warning (0.005 sec)
```

MariaDB [sge]>

## Calle

Calle(id\_cal, id\_loc2, calle, alt\_cal)  
 Calle(id\_cal, id\_loc2, calle, altura\_cal)

Calle		
Nombre	Tipo	Descripción
id_cal	INT	PK alias pk_ccal
id_loc2	INT	FK alias fk_cloc2
calle	VARCHAR(50)	
altura_cal	INT	Es el número de la calle. Por ejemp. <i>Alsina al 2100</i>

### alias de claves

pk\_ccal alias de id\_cal primary key entidad Calle  
 fk\_cloc2 alias de id\_cloc2 foreign key entidad Localidad

```
MariaDB [sge]> CREATE TABLE Calle(
-> id_cal INT,
-> id_loc2 INT,
-> calle VARCHAR(150),
-> altura_cal INT,
-> CONSTRAINT pk_ccal PRIMARY KEY (id_cal),
-> CONSTRAINT fk_cloc FOREIGN KEY (id_loc2) REFERENCES Localidad (id_loc));
Query OK, 0 rows affected, 1 warning (0.005 sec)
```

MariaDB [sge]>

## ParajeColonia

ParajeColonia(id\_par, id\_loc3, parCol)  
 ParajeColonia(id\_par, id\_loc3, parajeColonia)

ParajeColonia		
Nombre	Tipo	Descripción
id_par	INT	PK alias pk_cpc
id_loc3	INT	FK alias fk_cloc3
parajeColonia	VARCHAR(60)	

### alias de claves

pk\_cpc alias de id\_par primary key entidad ParajeColonia  
 fk\_cloc3 alias de id\_loc3 foreign key entidad Localidad

```
MariaDB [sge]> CREATE TABLE ParajeColonia(
-> id_par INT,
```

```

-> id_loc3 INT,
-> parajeColonia VARCHAR(150),
-> CONSTRAINT pk_cpc PRIMARY KEY (id_par),
-> CONSTRAINT fk_cloc3 FOREIGN KEY (id_loc3) REFERENCES Localidad(id_loc));
Query OK, 0 rows affected, 1 warning (0.005 sec)

```

MariaDB [sge]>

```

MariaDB [sge]> CREATE TABLE ParajeColonia(
-> id_par INT,
-> id_loc3 INT,
-> parajeColonia VARCHAR(150),
-> CONSTRAINT pk_cpc PRIMARY KEY (id_par),
-> CONSTRAINT fk_cloc3 FOREIGN KEY (id_loc3) REFERENCES Localidad(id_loc));
Query OK, 0 rows affected (0.23 sec)

```

MariaDB [sge]>

## Contacto

Contacto(id\_con, tipo, lugar)

Contacto		
Nombre	Tipo	Descripción
id_con	INT	PK alias pk_ccon
tipo	VARCHAR(7)	Móvil o Celular
lugar	VARCHAR(12)	Casa, Trabajo, Casa de Campo, etc

### alias de claves

pk\_ccon alias de id\_con primary key entidad Contacto

```

MariaDB [sge]> CREATE TABLE Contacto(
-> id_con INT,
-> tipo VARCHAR(7),
-> lugar VARCHAR(12),
-> CONSTRAINT pk_ccon PRIMARY KEY (id_con));
Query OK, 0 rows affected, 1 warning (0.004 sec)

```

MariaDB [sge]>

## Institución

Institucion(id\_inst, nom\_inst, nivel\_educ)  
 Institucion(id\_inst, institucion, nivel\_educ)

Institucion		
Nombre	Tipo	Descripción
id_inst	INT	PK alias pk_cinst
institucion	VARCHAR(50)	
nivel	VARCHAR(15)	Secundario, Técnico, Terciario, etc.

### alias de claves

pk\_cinst alias de id\_inst primary key entidad Institucion

```
MariaDB [sge]> CREATE TABLE Institucion(
-> id_inst INT,
-> institucion VARCHAR(150),
-> nivel VARCHAR(15),
-> CONSTRAINT pk_cinst PRIMARY KEY(id_inst));
Query OK, 0 rows affected, 1 warning (0.004 sec)
```

```
MariaDB [sge]>
```

### Título

```
Titulo(id_tit, denom_tit, resol_tit)
Titulo(id_tit, denominacion, resol_cre)
```

Titulo		
Nombre	Tipo	Descripción
id_tit	INT	PK alias pk_ctit
denominacion	VARCHAR(50)	
resol_cre	VARCHAR(15)	

#### alias de claves

pk\_ctit alias de id\_tit primary key entidad Título

```
MariaDB [sge]> CREATE TABLE Titulo(
-> id_tit INT,
-> denominacion VARCHAR(50),
-> resol_cre VARCHAR(15),
-> CONSTRAINT pk_ctit PRIMARY KEY (id_tit));
Query OK, 0 rows affected, 1 warning (0.004 sec)
```

```
MariaDB [sge]>
```

### Alumno

```
Alumno(id_alu, id_prov1, id_loc1, id_cal1, id_par1, app_alu, apm_alu, nombre_alu, tipo_doc,
numero_doc, genero, fecha_nac)
```

Alumno		
Nombre	Tipo	Descripción
id_alu	INT	PK alias pk_calu
id_prov1	INT	FK alias fk_cprov1
id_loc1	INT	FK alias fk_cloc1
id_cal1	INT	FK alias fk_ccal1
id_par1	INT	FK alias fk_cpar1
app_alu	VARCHAR(30)	Apellido Paterno del alumno
apm_alu	VARCHAR(30)	Apellido Materno
nombre_alu	VARCHAR(50)	
tipo_doc	VARCHAR(6)	
numero_doc	VARCHAR(12)	
genero	CHAR(1)	M - Masculino, F - Femenino
fecha_nac	DATETIME	Fecha de Nacimiento

#### alias de claves

pk\_calu alias de id\_alu primary key entidad Alumno  
fk\_cprov1 alias de id\_prov1 foreign key entidad Provincia  
fk\_cloc1 alias de id\_loc1 foreign key entidad Localidad  
fk\_ccal1 alias de id\_cal1 foreign key entidad Calle  
fk\_cpar1 alias de id\_par1 foreign key entidad ParajeColonia

```

MariaDB [sge]> CREATE TABLE Alumno(
  -> id_alu INT,
  -> id_prov1 INT,
  -> id_loc1 INT,
  -> id_cal1 INT,
  -> id_par1 INT,
  -> app_alu VARCHAR(50),
  -> apm_alu VARCHAR(50),
  -> nombre_alu VARCHAR(150),
  -> tipo_doc VARCHAR(6),
  -> numero_doc VARCHAR(12),
  -> genero CHAR(1),
  -> fecha_nac DATETIME,
  -> CONSTRAINT pk_calu PRIMARY KEY (id_alu),
  -> CONSTRAINT fk_cprov1 FOREIGN KEY (id_prov1) REFERENCES Provincia(id_prov),
  -> CONSTRAINT fk_cloc1 FOREIGN KEY (id_loc1) REFERENCES Localidad(id_loc),
  -> CONSTRAINT fk_ccal1 FOREIGN KEY (id_cal1) REFERENCES Calle(id_cal),
  -> CONSTRAINT fk_cpar1 FOREIGN KEY (id_par1) REFERENCES ParajeColonia(id_par));
Query OK, 0 rows affected, 1 warning (0.008 sec)

```

MariaDB [sge]>

## Profesor

Profesor(id\_prof, id\_prov2, id\_loc2, id\_cal2, id\_par2, app\_prof, apm\_prof, nom\_prof, tip\_doc, num\_doc, genero, f\_nac)

Profesor(id\_prof, id\_prov2, id\_loc2, id\_cal2, id\_par2, app\_prof, apm\_prof, nombre\_prof, tipo\_doc, numero\_doc, genero, fecha\_nac)

Profesor		
Nombre	Tipo	Descripción
id_prof	INT	PK alias pk_cprof
id_prov4	INT	FK alias fk_cprov4
id_loc4	INT	FK alias fk_cloc4
id_cal2	INT	FK alias fk_ccal2
id_par2	INT	FK alias fk_cpar2
app_prof	VARCHAR(30)	Apellido Paterno del Profesor
apm_prof	VARCHAR(30)	Apellido Materno
nombre_prof	VARCHAR(50)	
tipo_doc	VARCHAR(6)	
numero_doc	VARCHAR(12)	
genero	CHAR(1)	M - Masculino, F - Femenino
fecha_nac	DATETIME	Fecha de Nacimiento

### alias de claves

pk\_cprof alias de id\_prof primary key entidad Profesor  
 fk\_cprov4 alias de id\_prov4 foreign key entidad Provincia  
 fk\_cloc4 alias de id\_loc4 foreign key entidad Localidad  
 fk\_ccal2 alias de id\_cal2 foreign key entidad Calle  
 fk\_cpar2 alias de id\_par2 foreign key entidad ParajeColonia

```

MariaDB [sge]> CREATE TABLE Profesor(
  -> id_prof INT,
  -> id_prov4 INT,
  -> id_loc4 INT,
  -> id_cal2 INT,
  -> id_par2 INT,

```

```

-> app_prof VARCHAR(50),
-> apm_prof VARCHAR(50),
-> nombre_prof VARCHAR(150),
-> tipo_doc VARCHAR(6),
-> numero_doc VARCHAR(12),
-> genero CHAR(1),
-> fecha_nac DATETIME,
-> CONSTRAINT pk_cprof PRIMARY KEY (id_prof),
-> CONSTRAINT fk_cprov4 FOREIGN KEY (id_prov4) REFERENCES Provincia(id_prov),
-> CONSTRAINT fk_cloc4 FOREIGN KEY (id_loc4) REFERENCES Localidad(id_loc),
-> CONSTRAINT fk_ccal2 FOREIGN KEY (id_cal2) REFERENCES Calle(id_cal),
-> CONSTRAINT fk_cpar2 FOREIGN KEY (id_par2) REFERENCES ParajeColonia(id_par));
Query OK, 0 rows affected, 1 warning (0.007 sec)

```

MariaDB [sge]>

## Interrelaciones

### Calificación

Calificacion(id\_cal, id\_alu1, id\_ec1, ciclo\_lectivo, n1pt, n2pt, n3pt, n4pt, n5pt, n1st, n2st, n3st, n4st, n5st, n1tt, n2tt, n3tt, n4tt, n5tt, iedi, cfin)

Calificacion		
Nombre	Tipo	Descripción
id_cal	INT	PK alias pk_ccal
id_alu1	INT	FK alias fk_calu1
id_ec1	INT	FK alias fk_cec1
ciclo_lectivo	DATETIME	
n1pt	INT	Nota 1 del primer Trimestre
n2pt	INT	Nota 2 del primer Trimestre
n3pt	INT	Nota 3 del primer Trimestre
n4pt	INT	Nota 4 del primer Trimestre
n5pt	INT	Nota 5 del primer Trimestre
n1st	INT	Nota 1 del Segundo Trimestre
n2st	INT	Nota 2 del Segundo Trimestre
n3st	INT	Nota 3 del Segundo Trimestre
n4st	INT	Nota 4 del Segundo Trimestre
n5st	INT	Nota 5 del Segundo Trimestre
n1tt	INT	Nota 1 del Tercer Trimestre
n2tt	INT	Nota 2 del Tercer Trimestre
n3tt	INT	Nota 3 del Tercer Trimestre
n4tt	INT	Nota 4 del Tercer Trimestre
n5tt	INT	Nota 5 del Tercer Trimestre
iedi	INT	Instancia de Evaluación de Diciembre
cfin	INT	Calificación Final

#### alias de claves

pk\_ccal alias de id\_cal primary key tabla Calificación  
 fk\_calu1 alias de id\_alu1 foreign key entidad Alumno  
 fk\_cec1 alias de id\_ec1 foreign key entidad EspacioCurricular

```

MariaDB [sge]> CREATE TABLE Calificacion(
-> id_calif INT,
-> id_alu1 INT,
-> id_ec1 INT,
-> ciclo_lectivo DATETIME,
-> n1pt DECIMAL(4,2) UNSIGNED ZEROFILL,

```

```

-> n2pt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n3pt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n4pt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n5pt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n1st DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n2st DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n3st DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n4st DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n5st DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n1tt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n2tt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n3tt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n4tt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> n5tt DECIMAL(4,2) UNSIGNED ZEROFILL,
-> iedi DECIMAL(4,2) UNSIGNED ZEROFILL,
-> cfin DECIMAL(4,2) UNSIGNED ZEROFILL,
-> CONSTRAINT pk_ccalif PRIMARY KEY (id_calif),
-> CONSTRAINT fk_calu1 FOREIGN KEY (id_alu1) REFERENCES Alumno(id_alu),
-> CONSTRAINT fk_cec1 FOREIGN KEY (id_ec1) REFERENCES EspacioCurricular(id_ec));
Query OK, 0 rows affected, 1 warning (0.006 sec)

```

MariaDB [sge]>

### ImparteClasesDe

ImparteClasesDe(id\_impc, id\_prof1, id\_ec2, fecha\_inicio, fecha\_fin, sr)

ImparteClasesDe		
Nombre	Tipo	Descripción
id_impc	INT	PK alia pk_cimpc
id_prof1	INT	FK alias fk_cprof1
id_ec2	INT	FK alias fk_cec2
fecha_inicio	DATETIME	
fecha_fin	DATETIME	
sr	VARCHAR(9)	Titular, Interino, Suplente

#### alias de claves

pk\_cimpc alias de id\_impc primary key tabla ImparteClasesDe  
 fk\_cprof1 alias de id\_prf1 foreign key entidad Profesor  
 fk\_cec2 alias de id\_ec2 foreign key entidad EspacioCurricular

```

MariaDB [sge]> CREATE TABLE ImparteClasesDe(
-> id_impc INT,
-> id_prof1 INT,
-> id_ec2 INT,
-> fecha_inicio DATETIME,
-> fecha_fin DATETIME,
-> sr VARCHAR(9),
-> CONSTRAINT pk_cimpc PRIMARY KEY (id_impc),
-> CONSTRAINT fk_cprof1 FOREIGN KEY (id_prof1) REFERENCES Profesor(id_prof),
-> CONSTRAINT fk_cec2 FOREIGN KEY (id_ec2) REFERENCES EspacioCurricular(id_ec));
Query OK, 0 rows affected, 1 warning (0.006 sec)

```

MariaDB [sge]>



Es

Es(id\_es, id\_prof2, id\_tit1, fecha\_rec)

Es		
Nombre	Tipo	Descripción
id_es	INT	PK alias pk_ces
id_prof2	INT	FK alias fk_cprof2
id_tit1	INT	FK alias fk_ctit1
fecha_rec	DATETIME	Fecha de Recibido

## alias de claves

pk\_ces alias de id\_es primary key tabla Es  
 fk\_cprof2 alias de id\_prof2 foreign key entidad Profesor  
 fk\_ctit1 alias de id\_tit1 foreign key entidad Titulo

```
MariaDB [sge]> CREATE TABLE Es(
-> id_es INT,
-> id_prof2 INT,
-> id_tit1 INT,
-> fecha_rec DATETIME,
-> CONSTRAINT pk_ces PRIMARY KEY(id_es),
-> CONSTRAINT fk_cprof2 FOREIGN KEY (id_prof2) REFERENCES Profesor(id_prof),
-> CONSTRAINT fk_ctit1 FOREIGN KEY (id_tit1) REFERENCES Titulo(id_tit));
Query OK, 0 rows affected, 1 warning (0.006 sec)
```

MariaDB [sge]&gt;

OtorgadoPor

OtorgadoPor(id\_otor, id\_tit2, id\_inst1)

OtorgadoPor		
Nombre	Tipo	Descripción
id_otor	INT	PK alias pk_cotor
id_tit2	INT	FK alias fk_ctit2
id_inst1	INT	FK alias fk_cinst1

## alias de claves

pk\_cotor alias de id\_otor primary key tabla OtorgadoPor  
 fk\_ctit2 alias de id\_tit2 foreign key entidad Titulo  
 fk\_cinst1 alias de id\_inst1 foreign key entidad Institucion

```
MariaDB [sge]> CREATE TABLE OtorgadoPor(
-> id_otor INT,
-> id_tit2 INT,
-> id_inst1 INT,
-> CONSTRAINT pk_cotor PRIMARY KEY (id_otor),
-> CONSTRAINT fk_ctit2 FOREIGN KEY (id_tit2) REFERENCES Titulo(id_tit),
-> CONSTRAINT fk_cinst1 FOREIGN KEY (id_inst1) REFERENCES Institucion(id_inst));
Query OK, 0 rows affected, 1 warning (0.007 sec)
```

MariaDB [sge]&gt;

## TelAlu

TelAlu(id\_talu, id\_alu2, id\_con1)

TelAlu		
Nombre	Tipo	Descripción
id_talu	INT	PK alias pk_ctalu
id_alu2	INT	FK alias fk_calu2
id_con1	INT	FK alias fk_ccon1
numero	INT	

### alias de claves

pk\_ctalu alias de id\_talu primary key tabla TelAlu  
 fk\_calu2 alias de id\_alu2 foreign key entidad Alumno  
 fk\_ccon1 alias de id\_con1 foreign key entidad Contacto

```
MariaDB [sge]> CREATE TABLE TelAlu(
-> id_talu INT,
-> id_alu2 INT,
-> id_con1 INT,
-> numero INT,
-> CONSTRAINT pk_ctalu PRIMARY KEY (id_talu),
-> CONSTRAINT fk_calu2 FOREIGN KEY (id_alu2) REFERENCES Alumno(id_alu),
-> CONSTRAINT fk_ccon1 FOREIGN KEY (id_con1) REFERENCES Contacto(id_con));
Query OK, 0 rows affected, 1 warning (0.007 sec)
```

MariaDB [sge]>

## TelProf

TelProf(id\_tprof, id\_prof3, id\_con2)

Tel_prof		
Nombre	Tipo	Descripción
id_tprof	INT	PK alias pk_ctprof
id_prof3	INT	FK alias fk_cprof3
id_con2	INT	FK alias fk_ccon2
numero	INT	

### alias de claves

pk\_ctprof alias de id\_tprof primary key tabla TelProf  
 fk\_cprof3 alias de id\_prof3 foreign key entidad Profesor  
 fk\_ccon2 alias de id\_con2 foreign key entidad Contacto

```
MariaDB [sge]> CREATE TABLE TelProf(
-> id_tprof INT,
-> id_prof3 INT,
-> id_con2 INT,
-> numero INT,
-> CONSTRAINT pk_ctprof PRIMARY KEY (id_tprof),
-> CONSTRAINT fk_cprof3 FOREIGN KEY (id_prof3) REFERENCES Profesor(id_prof),
-> CONSTRAINT fk_ccon2 FOREIGN KEY (id_con2) REFERENCES Contacto(id_con));
Query OK, 0 rows affected, 1 warning (0.007 sec)
```

MariaDB [sge]>

